

# Technical Assessment: ERS

---

Author: Matthew W. Noble, DPhil [[About the Author](#)]

Date of Submission: 2018 · January · 08

## Preamble

This technical assessment was performed by Matthew W. Noble for the Pricing Analyst role at ERS. The deadline for submission was 07:00 on 2018/01/08. The [Brief](#), [What to submit](#), and [Data Dictionary](#) were provided by ERS to explain the task. In addition to this information, the data to be used was provided as a .csv file named:

**ERS\_Technical\_Assessment\_data.csv**

and a sample .csv submission file was provided, named:

**Sample\_Submission.csv**

All materials provided and used throughout are available in the directory:

**...\CodingChallenges\ERS**

## Brief

Evaluate the relative risk of a number of regions. Use the provided vehicle risk data with a selection of area related features attached to evaluate the relative risk of each region. You must assign a rating (1-99, low to high) for each region that is representative of its relative risk.

The dataset comprises of 32 columns, including 28 of area-related features that you may use to analyse relative area risk.

You may analyse the data any way you like. However, assessment will consider both the technical quality of your work and your ability to present the results. You can assume you will be presenting to an audience familiar with statistical methods and the insurance context (senior actuary or senior underwriter). You should be prepared to answer technical questions on the methods that you have elected to use.

Please annotate your code/analysis and ensure that your presentation includes the following:

1. Brief commentary on rationale for the method(s) employed<sup>1</sup>
2. Analysis of strengths and weaknesses of your approach
3. Brief description of any evaluation metric(s) that you used

---

<sup>1</sup> You can assume your audience understands canonical modelling techniques — please do not present a detailed explanation of a method.

## What to submit:

Candidates must submit three exhibits:

- 1) **Short presentation**, (15 min) which you will present as part of the assessment. You should be prepared to explain and defend your work during a 15 min Q&A session following the presentation. Please provide as either PowerPoint/pdf/RMarkdown/html etc.
- 2) **CSV file with each of the regions and resulting 1-99 scores**. Please give the regions a rating between 1 and 99 related to their predicted risk level (with 99 being the highest risk regions). We have supplied a template for the submission exhibit.<sup>2</sup>
- 3) **Supporting information**, ideally as a single zip file. Please provide sufficient code, model files and/or documentation of your analysis in a legible format (html, pynb, R, excel etc.) so that your work could be repeated by an appropriately trained analyst. Please ensure that any accompanying visualisations and annotations render correctly and can be viewed after emailing.

---

<sup>2</sup> See file “Sample\_Submission.csv”.

# Data Dictionary

## Observation details

Region	An identifier for each region
Frequency	Rate of Claims within a given region (Number of Claims/Exposure)
Exposure (EVY)	Measure of exposure within the Region in earned vehicle years; Measure of the experience within the region (how much time*number of vehicles the data has been recorded for)
Non_Area_Related_Veh_Risk	Measure of Non-Area Related risk for the given region; a measure of risk that is not related to the Area (based on other factors such as types of car, age of drivers etc.)

## Area-related features

Location_Type	Type of Location
Local_Authority_Code	Code representing different Local Authorities
Density	Population Density
Traffic_Flow	Average Traffic Flow within the region
Distance_Travelled	Average Distance travelled within the Region
Averaged_Density	Population Density (Smoothed)
Distance_Travelled_2	Average Distance travelled within the Region (Smoothed)
Local_Crimes	Number of Local Crimes
Young	Proportion of the population that is 'Young'
Mid	Proportion of the population that is 'Middle Aged'
Old	Proportion of the population that is 'Old'
Points_Per_License_Avg	Average points per License (Smoothed)
Avg_Drivers	Average number of Drivers within the Region (Smoothed)
Avg_Pts	Average Number of Penalty Points (Smoothed)
Points_Per_License	Average points per License
Drivers	Average number of Drivers within the Region
Pts	Average Number of Penalty Points
Local_Schools	Average number of Local Schools
Local_CMCs	Average number of Claims Management Companies
Nearest_CMC	Average distance to the nearest Claims Management Company
Nearest_School	Average distance to the nearest School
Nearest_Incidents	Average distance to the nearest Incident
Nearest_Crimes	Average distance to the nearest Crime
Local_Incidents_Sev1	Number of High Severity Incidents
Local_Incidents_Sev2	Number of Medium Severity Incidents
Local_Incidents_Sev3	Number of Low Severity Incidents
Incident_Severity_Ratio1	Ratio of the number of High to Low Severity Incidents
Incident_Severity_Ratio2	Ratio of the number of High to Low Severity Incidents (2)

# Data Analysis

Printing the `.head(10)` of each of the columns:

Region 1 0.073234 2 0.059753 3 0.000000 4 0.000000 5 0.037753 6 0.074187 7 0.000000 8 0.000000 9 0.000000 10 0.000000 Name: Frequency, dtype: float64	Region 1 13.654795 2 33.471233 3 27.657534 4 4.397260 5 26.487671 6 13.479452 7 11.802740 8 9.660274 9 23.449315 10 32.542466 Name: Exposure (EVY), dtype: float64	Region 1 0.051595 2 0.030824 3 0.035585 4 0.023894 5 0.029252 6 0.040134 7 0.040027 8 0.039755 9 0.027125 10 0.032035 Name: Non_Area_Related_Veh_Risk, dtype: float64
Region 1 Large urban area 2 Large urban area 3 Large urban area 4 Large urban area 5 Large urban area 6 Large urban area 7 Large urban area 8 Large urban area 9 Large urban area 10 Accessible small town Name: Location_Type, dtype: object	Region 1 338 2 338 3 338 4 338 5 338 6 338 7 338 8 338 9 338 10 339 Name: Local_Authority_Code, dtype: int64	Region 1 11.870000 2 11.870000 3 11.870000 4 11.870000 5 11.870000 6 11.870000 7 11.870000 8 11.870000 9 11.494291 10 0.454134 Name: Density, dtype: float64
Region 1 1314.000000 2 1314.000000 3 1314.000000 4 1314.000000 5 1314.000000 6 1314.000000 7 1314.000000 8 1314.000000 9 1362.861818 10 2798.659218 Name: Traffic_Flow, dtype: float64	Region 1 4.715810 2 4.715810 3 4.715810 4 4.715810 5 4.715810 6 4.715810 7 4.715810 8 4.715810 9 4.852145 10 8.858346 Name: Distance_Travelled, dtype: float64	Region 1 11.639188 2 11.639188 3 11.639188 4 11.725474 5 11.725474 6 11.725474 7 11.725474 8 11.725474 9 10.223148 10 10.223148 Name: Averaged_Density, dtype: float64
Region 1 4.735272 2 4.735272 3 4.735272 4 4.727951 5 4.727951 6 4.727951 7 4.727951 8 4.727951 9 4.888633 10 5.614635 Name: Distance_Travelled_2, dtype: float64	Region 1 NaN 2 NaN 3 NaN 4 NaN 5 NaN 6 NaN 7 NaN 8 NaN 9 NaN 10 NaN Name: Local_Crimes, dtype: float64	Region 1 0.297859 2 0.297859 3 0.297859 4 0.297859 5 0.297859 6 0.297859 7 0.297859 8 0.297859 9 0.293943 10 0.178874 Name: Young, dtype: float64
Region 1 0.596964 2 0.596964 3 0.596964 4 0.596964 5 0.596964 6 0.596964 7 0.596964 8 0.596964 9 0.600482 10 0.703831 Name: Mid, dtype: float64	Region 1 0.105177 2 0.105177 3 0.105177 4 0.105177 5 0.105177 6 0.105177 7 0.105177 8 0.105177 9 0.105576 10 0.117295 Name: Old, dtype: float64	Region 1 0.385406 2 0.385406 3 0.385406 4 0.387316 5 0.387316 6 0.387316 7 0.387316 8 0.387316 9 0.385936 10 0.385936 Name: Points_Per_License_Avg, dtype: float64
Region 1 4.503306e+09 2 4.503306e+09 3 4.503306e+09 4 4.352629e+09	Region 1 1.735603e+09 2 1.735603e+09 3 1.735603e+09 4 1.685844e+09	Region 1 0.381379 2 0.381379 3 0.381379 4 0.455668

5 4.352629e+09 6 4.352629e+09 7 4.352629e+09 8 4.352629e+09 9 4.063142e+09 10 4.063142e+09 Name: Avg_Drivers, dtype: float64	5 1.685844e+09 6 1.685844e+09 7 1.685844e+09 8 1.685844e+09 9 1.568113e+09 10 1.568113e+09 Name: Avg_Pts, dtype: float64	5 0.455668 6 0.455668 7 0.455668 8 0.455668 9 0.430636 10 0.430636 Name: Points_Per_License, dtype: float64
Region 1 13619 2 13619 3 13619 4 10613 5 10613 6 10613 7 10613 8 10613 9 18057 10 18057 Name: Drivers, dtype: int64	Region 1 5194 2 5194 3 5194 4 4836 5 4836 6 4836 7 4836 8 4836 9 7776 10 7776 Name: Pts, dtype: int64	Region 1 31.229665 2 30.393443 3 20.801802 4 24.772059 5 28.916279 6 24.385965 7 18.056338 8 20.010638 9 8.134545 10 4.770950 Name: Local_Schools, dtype: float64
Region 1 0.0 2 0.0 3 0.0 4 0.0 5 0.0 6 0.0 7 0.0 8 0.0 9 0.0 10 0.0 Name: Local_CMCs, dtype: float64	Region 1 92510.53574 2 91301.51103 3 89885.89469 4 93174.01919 5 92332.31336 6 91829.33639 7 92414.77910 8 92663.55376 9 89711.34458 10 84468.03051 Name: Nearest_CMC, dtype: float64	Region 1 265.038480 2 329.699334 3 528.182239 4 506.163139 5 372.889342 6 410.321541 7 341.666668 8 466.276593 9 646.086135 10 631.437361 Name: Nearest_School, dtype: float64
Region 1 1095.857770 2 1908.613982 3 3014.955096 4 1896.529820 5 1040.038110 6 2611.003055 7 2223.648806 8 2340.519333 9 4516.650995 10 5152.652589 Name: Nearest_Incidents, dtype: float64	Region 1 NaN 2 NaN 3 NaN 4 NaN 5 NaN 6 NaN 7 NaN 8 NaN 9 NaN 10 NaN Name: Nearest_Crimes, dtype: float64	Region 1 9.076555 2 9.118852 3 10.797297 4 9.044118 5 9.018605 6 9.000000 7 9.000000 8 9.021277 9 10.963636 10 11.000000 Name: Local_Incidents_Sev1, dtype: float64
Region 1 105.976077 2 108.057377 3 111.711712 4 103.080882 5 105.232558 6 105.315790 7 101.577465 8 102.000000 9 107.109091 10 120.474860 Name: Local_Incidents_Sev2, dtype: float64	Region 1 357.832536 2 360.200820 3 354.977477 4 350.492647 5 356.511628 6 352.771930 7 340.795775 8 343.638298 9 328.538182 10 348.217877 Name: Local_Incidents_Sev3, dtype: float64	Region 1 0.019206 2 0.019101 3 0.022613 4 0.019554 5 0.019160 6 0.019270 7 0.019941 8 0.019844 9 0.024570 10 0.022942 Name: Incident_Severity_Ratio1, dtype: float64
Region 1 0.243320 2 0.245450 3 0.256575 4 0.242389 5 0.242697 6 0.244741 7 0.244990 8 0.244206 9 0.264432 10 0.274152 Name: Incident_Severity_Ratio2, dtype: float64		

Generating the correlation matrix:

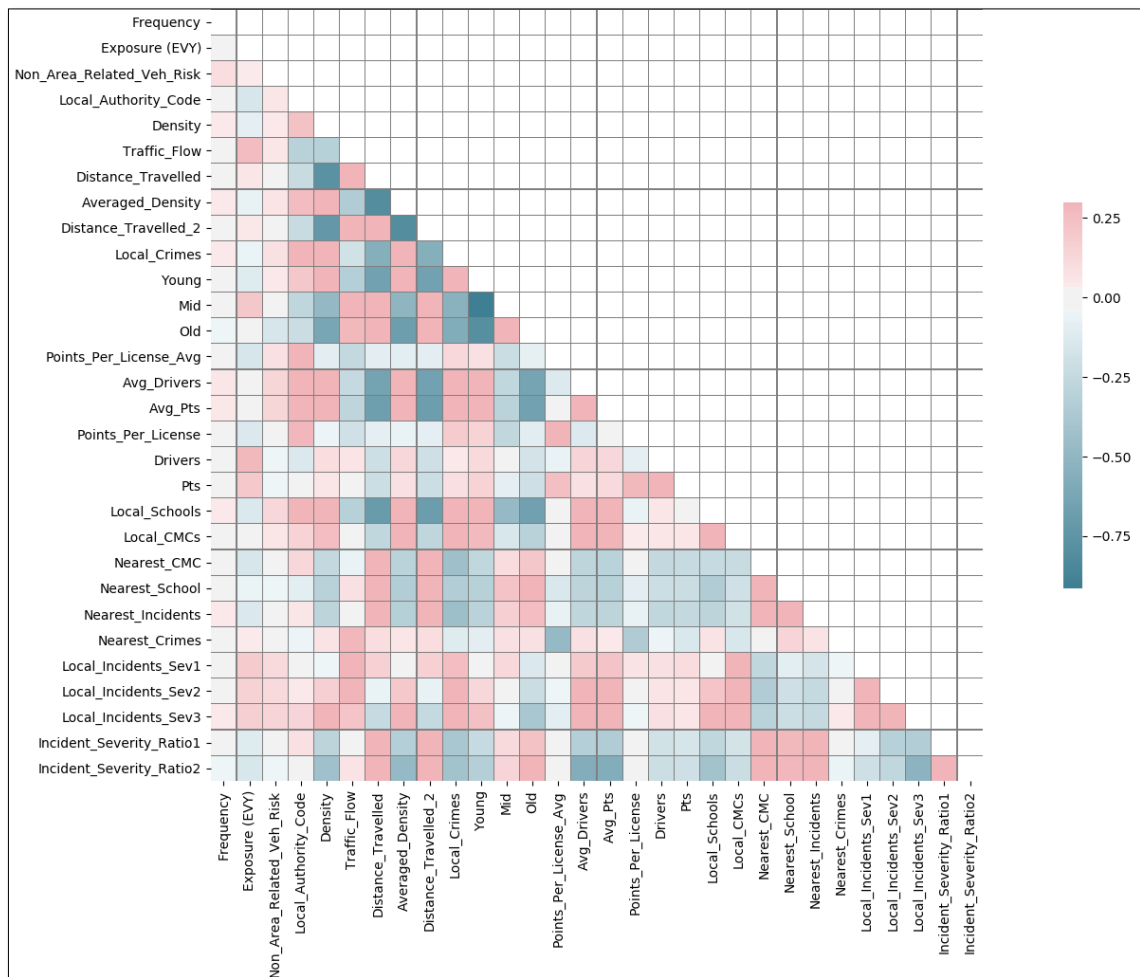


Figure 1: Correlation matrix for all of the numerical fields.

Printing the `.info()` of the DataFrame:

```
<class 'pandas.core.frame.DataFrame'>

Int64Index: 8995 entries, 1 to 8995

Data columns (total 31 columns):
Frequency                8995 non-null float64
Exposure (EVY)           8995 non-null float64
Non_Area_Related_Veh_Risk 8995 non-null float64
Location_Type            8995 non-null object
Local_Authority_Code      8995 non-null int64
Density                  8995 non-null float64
Traffic_Flow              8995 non-null float64
Distance_Travelled        8995 non-null float64
Averaged_Density          8995 non-null float64
Distance_Travelled_2      8995 non-null float64
Local_Crimes              8076 non-null float64
Young                    8995 non-null float64
Mid                      8995 non-null float64
Old                      8995 non-null float64
Points_Per_License_Avg    8995 non-null float64
Avg_Drivers               8995 non-null float64
Avg_Pts                   8995 non-null float64
Points_Per_License        8995 non-null float64
Drivers                   8995 non-null int64
Pts                       8995 non-null int64
Local_Schools             8995 non-null float64
Local_CMCs                8995 non-null float64
Nearest_CMC               8995 non-null float64
Nearest_School            8995 non-null float64
Nearest_Incidents         8995 non-null float64
Nearest_Crimes            8076 non-null float64
Local_Incidents_Sev1      8995 non-null float64
Local_Incidents_Sev2      8995 non-null float64
Local_Incidents_Sev3      8995 non-null float64
Incident_Severity_Ratio1   8994 non-null float64
Incident_Severity_Ratio2   8994 non-null float64

dtypes: float64(27), int64(3), object(1)

memory usage: 2.2+ MB

None
```



Printing the `.describe()` of the columns:

count      8995.000000 mean        0.028664 std         0.050857 min         0.000000 25%         0.000000 50%         0.021409 75%         0.042160 max         3.041667 Name: Frequency, dtype: float64	count      8995.000000 mean        53.418893 std         46.063739 min         0.024658 25%         22.547945 50%         43.430137 75%         72.052055 max        717.043835 Name: Exposure (EVY), dtype: float64	count      8995.000000 mean        0.028617 std         0.006235 min         0.004053 25%         0.024827 50%         0.028214 75%         0.031771 max         0.108312 Name: Non_Area_Related_Veh_Risk, dtype: float64
count      8995 unique       18 top    Urban city and town freq       3233 Name: Location_Type, dtype: object	count      8995.000000 mean        203.059255 std         116.543219 min         1.000000 25%         87.000000 50%        234.000000 75%        298.000000 max        380.000000 Name: Local_Authority_Code, dtype: float64	count      8995.000000 mean        20.951530 std         27.788004 min         0.090000 25%         2.300000 50%        10.900000 75%        30.400000 max        222.500000 Name: Density, dtype: float64
count      8995.000000 mean        4520.276012 std         4082.386467 min        130.328767 25%       1427.000000 50%       2640.093023 75%       7023.000000 max      15001.000000 Name: Traffic_Flow, dtype: float64	count      8995.000000 mean         5.934158 std          2.117191 min         0.760321 25%         4.238439 50%         5.930679 75%         7.635787 max         11.291168 Name: Distance_Travelled, dtype: float64	count      8995.000000 mean        20.695282 std         23.434869 min         0.090000 25%         2.770000 50%        13.415023 75%        30.420084 max        125.985899 Name: Averaged_Density, dtype: float64
count      8995.000000 mean         5.828959 std          2.071059 min         0.791770 25%         4.162229 50%         5.763227 75%         7.407108 max         11.291168 Name: Distance_Travelled_2, dtype: float64	count      8076.000000 mean        783.055966 std         742.687650 min         0.666667 25%        212.251489 50%        566.009427 75%       1118.286875 max       4435.114035 Name: Local_Crimes, dtype: float64	count      8995.000000 mean         0.219562 std          0.051975 min         0.140163 25%         0.179334 50%         0.203184 75%         0.250932 max         0.401254 Name: Young, dtype: float64
count      8995.000000 mean         0.680077 std          0.037085 min         0.545403 25%         0.665105 50%         0.688589 75%         0.706099 max         0.754181 Name: Mid, dtype: float64	count      8995.000000 mean         0.129887 std          0.029027 min         0.049409 25%         0.111566 50%         0.129427 75%         0.149674 max         0.235469 Name: Old, dtype: float64	count      8995.000000 mean         0.304938 std          0.064245 min         0.085989 25%         0.267553 50%         0.297476 75%         0.332714 max         0.571693 Name: Points_Per_License_Avg, dtype: float64
count      8.995000e+03 mean      1.692946e+10 std       1.984901e+10 min       5.375230e+06 25%       4.873011e+09 50%       1.047710e+10 75%       1.907769e+10 max       8.337202e+10 Name: Avg_Drivers, dtype: float64	count      8.995000e+03 mean      5.006738e+09 std       5.352610e+09 min       9.350906e+05 25%       1.387760e+09 50%       3.209109e+09 75%       6.539716e+09 max       2.201732e+10 Name: Avg_Pts, dtype: float64	count      8995.000000 mean      NaN std      NaN min       0.000000 25%       0.257017 50%       0.297646 75%       0.346812 max       Inf Name: Points_Per_License, dtype: float64
count      8995.000000 mean      18100.406782 std       10392.019105 min       0.000000 25%      10970.000000 50%      17186.000000 75%      23779.000000 max      83195.000000 Name: Drivers, dtype: float64	count      8995.000000 mean      5479.610450 std       3319.290799 min       0.000000 25%      3188.000000 50%      5054.000000 75%      7106.000000 max      25351.000000 Name: Pts, dtype: float64	count      8995.000000 mean      19.524790 std       24.797306 min       0.000000 25%       3.991331 50%      11.541667 75%      25.011430 max      149.000000 Name: Local_Schools, dtype: float64
count      8995.000000 mean       20.324137 std        29.858313 min        0.000000	count      8995.000000 mean      11202.220560 std       24657.107516 min       64.740482	count      8995.000000 mean      837.332549 std       972.181168 min       0.000000

25% 2.594828 50% 10.382353 75% 26.583341 max 208.548837 Name: Local_CMCs, dtype: float64	25% 1534.105676 50% 3664.657559 75% 10464.892175 max 441613.290100 Name: Nearest_CMC, dtype: float64	25% 392.948464 50% 553.347347 75% 938.226576 max 46747.374840 Name: Nearest_School, dtype: float64
count 8995.000000 mean 4442.679273 std 7557.616124 min 255.792433 25% 1538.527405 50% 2441.146737 75% 4774.950189 max 239210.402700 Name: Nearest_Incidents, dtype: float64	count 8076.000000 mean 15940.251645 std 22240.829822 min 666.881605 25% 3439.899580 50% 7337.108007 75% 18914.223595 max 267240.086400 Name: Nearest_Crimes, dtype: float64	count 8995.000000 mean 22.070543 std 14.543237 min 0.000000 25% 11.033231 50% 19.452128 75% 29.780088 max 94.231250 Name: Local_Incidents_Sev1, dtype: float64
count 8995.000000 mean 306.645092 std 214.542391 min 0.000000 25% 148.559063 50% 265.359060 75% 426.077506 max 1494.967391 Name: Local_Incidents_Sev2, dtype: float64	count 8995.000000 mean 2510.883463 std 2286.091082 min 0.000000 25% 963.833442 50% 1979.379310 75% 3196.966347 max 19024.195650 Name: Local_Incidents_Sev3, dtype: float64	count 8994.000000 mean 0.011575 std 0.012880 min 0.000000 25% 0.005861 50% 0.009025 75% 0.013061 max 0.538760 Name: Incident_Severity_Ratio1, dtype: float64
count 8994.000000 mean 0.138577 std 0.051668 min 0.000000 25% 0.109229 50% 0.134313 75% 0.162543 max 0.709302 Name: Incident_Severity_Ratio2, dtype: float64		

The **Location\_Type** column was a category column, looking at the unique counts of the categories via a `.crosstab()` implementation:

col_0	count
Location_Type	
Accessible rural area	129
Accessible small town	81
Large urban area	297
Other urban area	224
Remote rural area	57
Remote small town	24
Rural hamlet and isolated dwellings	357
Rural hamlet and isolated dwellings in a sparse setting	106
Rural town and fringe	799
Rural town and fringe in a sparse setting	77
Rural village	551
Rural village in a sparse setting	73
Urban city and town	3233
Urban city and town in a sparse setting	26
Urban major conurbation	2612
Urban minor conurbation	239
Very remote rural area	95
Very remote small town	15

## Understanding and Inspecting the Data

### Observation Details

The **Frequency** field is the: *measure of the Rate of Claims within a given region (Number of Claims/Exposure)*. **This will probably be important.** The higher this rate, the more likely a region is to make a claim.

Possible errors: There is a frequency of 3.041667 in Region 5139. This is the only point above 1.0 and since this is a rate, I believe this to be incorrect. I have changed this to the 2<sup>nd</sup> highest value of: 0.910224. This was done in Excel as it was faster.

The **Exposure (EVY)** field is the: *Measure of exposure within the Region in earned vehicle years; Measure of the experience within the region (how much time\*number of vehicles the data has been recorded for).* **This is probably not important** as I will include the **Frequency** field and the **Exposure** and **Frequency** fields are not independent.

The **Non\_Area\_Related\_Veh\_Risk** field is the: *Measure of Non-Area Related risk for the given region ; a measure of risk that is not related to the Area (based on other factors such as types of car, age of drivers etc.).* **This will probably be important.** The higher this rate, the more likely a region is to make a claim. This is independent of the area-related features, which is good.

## Area-related features

The **Location\_Type** field is the: *Type of Location.* **This is probably not important.** It will no doubt be linked to other things in the area-related features, but is itself not important.

The **Local\_Authority\_Code** field is the: *Code representing different Local Authorities.* **This is probably not important.** It will no doubt be linked to other things in the area-related features, but is itself not important.

The **Density** is the: *Population Density.* **This is probably not important.** The **Averaged\_Density** field will be used instead as it is smoothed.

The **Traffic\_Flow** field is the: *Average Traffic Flow within the region.* **This will probably be important.** The higher the traffic flow within a region, the more likely an accident is to happen.

The **Distance\_Travelled** field is the: *Average Distance travelled within the Region.* **This is probably not important.** The **Distance\_Travelled\_2** field will be used instead as it is smoothed.

The **Averaged\_Density** field is the: *Population Density (Smoothed).* **This will probably be important.** This will allow metrics to be calculated per population.

The **Distance\_Travelled\_2** field is the: *Average Distance travelled within the Region (Smoothed).* **This will probably be important.** My initial thoughts were that the longer the distance travelled, the more time spent inside a vehicle, and therefore the more likely to have an incident. Upon reflection, the more time spent inside of a vehicle, the more likely they are

going to transition to a motorway or be on a long journey, and hence not be continuously stopping and starting. Therefore the larger the distance, the less likely of an incident.

The **Local\_Crimes** field is the: *Number of Local Crimes*. **This is probably not important**. It will no doubt be linked to other things in the area-related features, but is itself not important. This investigation is looking at vehicle risk, therefore unless there is a way to separate out the vehicle crimes from all crimes (which I do not believe there is), this field is meaningless. Your risk of being burgled or mugged and your risk of being rear-ended are not related.

The **Young** field is the: *Proportion of the population that is 'Young'*. **This will probably be important**. Young drivers who have just passed their test do not yet have road experience and are more likely to be impulsive. Increased risk relative to the **Mid** field.

The **Mid** field is the: *Proportion of the population that is 'Middle Aged'*. **This will probably be important**. Mid drivers have road experience and are less likely to be impulsive. Decreased risk relative to both the **Young** and **Old** fields.

The **Old** field is the: *Proportion of the population that is 'Old'*. **This will probably be important**. Old drivers have the most road experience but have become complacent in their old age and are becoming sensory impaired. Increased risk relative to the **Mid** field.

The **Points\_Per\_License\_Avg** field is the: *Average points per License (Smoothed)*. **This will probably be important**. This is a measure of whether the drivers in that region obey the laws of the road or not.

The **Avg\_Drivers** field is the: *Average number of Drivers within the Region (Smoothed)*. **This will probably be important**. More drivers in a region means more likelihood of an incident occurring. This will also allow metrics to be calculated per driver.

The **Avg\_Pts** field is the: *Average Number of Penalty Points (Smoothed)*. **This is probably not important**. This is a smoothed value, which makes it better than the **Pts** field, but it is still an absolute. The **Points\_Per\_License\_Avg** will be used instead.

The **Points\_Per\_License** field is the: *Average points per License*. **This is probably not important**. The **Points\_Per\_License\_Avg** field will be used instead as it is smoothed.

Possible errors: Two values, for regions 5083 and 2330 were labelled as having Inf values. This is of course impossible. I believe this is linked to the **Drivers** field for the two regions being 0. If this is the average points per licence and a “licence” is defined as a “driver” then dividing by 0 would explode this to Inf. To correct the two entries, the data was filtered by the **Location\_Type** column such that only the Urban Major Conurbation data was sampled. Then going by the **Density** column, the values  $\pm 2.5\%$  were selected and the median of the **Points\_Per\_Licence** column within these two filters were taken (median rather than average

to defend against outliers). This value then replaced the incorrect Inf value. For region 5083, this meant a new value of 0.29938443 and for region 2330, a new value of 0.309702752.

The **Drivers** field is the: *Average number of Drivers within the Region.* **This is probably not important.** The **Avg\_Drivers** field will be used instead as it is smoothed.

The **Pts** field is the: *Average Number of Penalty Points.* **This is probably not important.** The **Points\_Per\_License** field is applicable to drivers as it is not an absolute.

The **Local\_Schools** field is the: *Average number of Local Schools.* **This is probably not important.** It will no doubt be linked to other things in the area-related features, but is itself not important.

The **Local\_CMCs** field is the: *Average number of Claims Management Companies.* **This is probably not important.** It will no doubt be linked to other things in the area-related features, but is itself not important. This may be used for marketing, knowing that there is more or less competition in a given area.

The **Nearest\_CMC** field is the: *Average distance to the nearest Claims Management Company.* **This is probably not important.** It will no doubt be linked to other things in the area-related features, but is itself not important. This may be used for marketing, knowing that there is more or less competition in a given area.

The **Nearest\_School** field is the: *Average distance to the nearest School.* **This is probably not important.** It will no doubt be linked to other things in the area-related features, but is itself not important.

The **Nearest\_Incidents** field is the: *Average distance to the nearest Incident.* **This may be important.**

The **Nearest\_Crimes** field is the: *Average distance to the nearest Crime.* **This is probably not important.** It will no doubt be linked to other things in the area-related features, but is itself not important.

The **Local\_Incidents\_Sev1** field is the: *Number of High Severity Incidents.* **This will probably be important.** The higher this number, the more incidents in the area. This is an absolute number, so dividing by the **Avg\_Drivers** field will reduce this to the number of high severity incidents per driver.

The **Local\_Incidents\_Sev2** field is the: *Number of Medium Severity Incidents.* **This will probably be important.** The higher this number, the more incidents in the area. This is an absolute number, so dividing by the **Avg\_Drivers** field will reduce this to the number of medium severity incidents per driver.

The **Local\_Incidents\_Sev3** field is the: *Number of Low Severity Incidents*. **This will probably be important**. The higher this number, the more incidents in the area. This is an absolute number, so dividing by the **Avg\_Drivers** field will reduce this to the number of low severity incidents per driver.

(!) I will sum the high, medium, and low severity incidents together. Ultimately a claim is still a claim that we would have to pay out over, regardless as to whether it was replacing a rear panel or a new car.

The **Incident\_Severity\_Ratio1** field is the: *Ratio of the number of High to Low Severity Incidents*. **This is probably not important**. It will no doubt be linked to other things in the area-related features, but is itself not important.

The **Incident\_Severity\_Ratio2** field is the: *Ratio of the number of High to Low Severity Incidents (2)*. **This is probably not important**. It will no doubt be linked to other things in the area-related features, but is itself not important.

## Building the model

Upon consideration after EDA, I identified the following 6 important fields:

- Frequency
- Non\_Area\_Related\_Veh\_Risk
- Number\_Of\_Claims\_Per\_Region\_Per\_Driver
  - = (Frequency) \* (Exposure (EVY)) / (Avg\_Drivers)
- Total\_Local\_Incidents\_Per\_Driver
  - = 
$$\frac{(\text{Local\_Incidents\_Sev1} + \text{Local\_Incidents\_Sev2} + \text{Local\_Incidents\_Sev3})}{(\text{Avg\_Drivers})}$$
- Traffic\_Flow
- Points\_Per\_Licence\_Avg

In order to combine the fields, I chose to scale them first via [linear interpolation](#), using the formula:

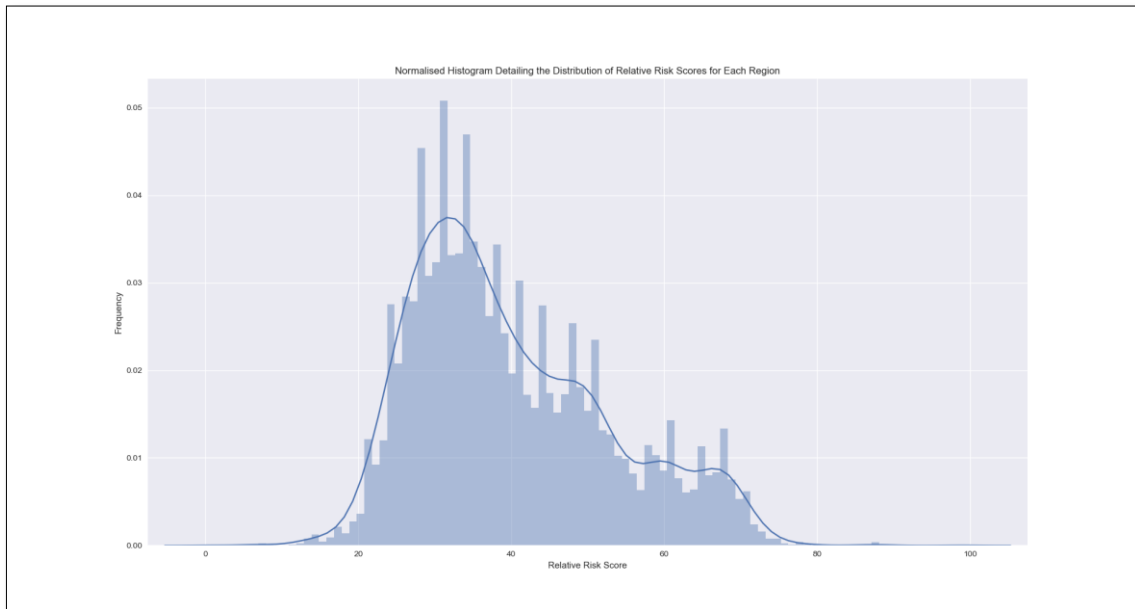
$$y = y_0 + (x - x_0) \frac{y_1 - y_0}{x_1 - x_0}$$

The idea being that I want to take a value,  $x$  which lies in the original range,  $x_0 \leq x \leq x_1$  and transform it into the new value,  $y$  which lies in the new range,  $y_0 \leq y \leq y_1$ . The new range was chosen to be 1 - 99 as that was the relative risk score value asked for. The values at the beginning and end of the range, would be scaled to 1 and 99 respectively and were chosen to be the minimum and maximum of the DataFrame column. However the values in between would be scaled relative to their distribution within that range.

To calculate a final risk score, the individual scaled scores from the 6 fields named above were all summed together and that sum was then scaled a final time. The distribution of relative risk scores is plotted as a normalised histogram in **figure 2**.

The Regions and the Scores were exported to a **.csv** file as requested with the name:

**TechnicalAssessmentERS\_Submission\_M\_W\_Noble.csv**



**Figure 2:** Normalised histogram detailing the distribution of relative risk scores for each region.

## Analysis and Comments

### Method

My method of linear interpolation goes some way to protect the scaling from outliers and seemed more intuitive than uniformly distributing via quantiles; it allowed for the easy combination of fields which possess values orders of magnitude apart; and finally, it allowed the regions of a given field to be easily scaled relative to each other within that field.

## Weightings

It would of course be possible to weight the fields in order to preference one over another, e.g.:

Individual Score	Weighting	Weighted Total Score
90	15%	75.25
85	15%	
70	70%	

but without knowing more about the “business understanding” I’m not sure if this would create a better model or simply reinforce my own confirmation bias. I therefore assumed all weights were equal in the process of generating the final risk score.



## About the Author

Hello, World! If you wish to contact me, please select from the following options:

### E-Mail Address and Personal Website

E-Mail: [Matthew.William.Noble@gmail.com](mailto:Matthew.William.Noble@gmail.com)  
Personal website: <http://matthewnoble.info/index.html>

### CV and Coding Portfolio

CV (pdf): <https://drive.google.com/open?id=1d6mdd39LRKL2DLfNmaY6xiOIIBJm5653>  
CV (online): <http://matthewnoble.info/MyCV.html>  
GitHub: <https://github.com/MatthewWilliamNoble/CodingPortfolio>

### Social Media

Facebook: <https://www.facebook.com/matthew.w.noble>  
LinkedIn: <https://www.linkedin.com/in/matthew-w-noble/>  
Twitter: [https://twitter.com/Matthew\\_W\\_Noble](https://twitter.com/Matthew_W_Noble)  
Instagram: <https://www.instagram.com/matthew.w.noble/>  
Kaggle: <https://www.kaggle.com/@matthew104>  
HackerEarth: <https://www.hackerearth.com/@matthew104>  
HackerRank: [https://www.hackerrank.com/matthew\\_noble\\_11](https://www.hackerrank.com/matthew_noble_11)

